Step 1: Set Up the Database Connection

Install `pymysql`: `pipenv install pymysql`

1.

Update `settings.py` (inside the project folder):

```
import pymysql
pymysql.install_as_MySQLdb()
```

2.

Configure `settings.py`: Locate `DATABASES` in the file and update it:

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'your_database_name',
        'USER': 'your_mysql_username',
        'PASSWORD': 'your_mysql_password',
        'HOST': '127.0.0.1',
        'PORT': '3306',
    }
}
```

3.

Run Migrations to test the connection:

```
pipenv run python manage.py migrate
```

4.

Step 2: Create a Django App

Create App:

```
pipenv run python manage.py startapp contest_app
```

1.

Add App to `INSTALLED_APPS` in `settings.py`:
python

CopyEdit

```
INSTALLED_APPS = [
    # Other apps
    'contest_app',
]
```

2.

Step 3: Define Models

Edit `models.py` in the `contest_app` folder:

```
from django.db import models

class Student(models.Model):
    name = models.CharField(max_length=128)
    email = models.EmailField(unique=True)
    registration_date = models.DateTimeField(auto_now_add=True)

    def __str__(self):
        return self.name

class Problem(models.Model):
    title = models.CharField(max_length=128)
    description = models.TextField()
    difficulty = models.CharField(max_length=50)

    def __str__(self):
        return self.title

class Solution(models.Model):
    student = models.ForeignKey(Student, on_delete=models.CASCADE)
    problem = models.ForeignKey(Problem, on_delete=models.CASCADE)
    solved_on = models.DateTimeField(auto_now_add=True)

    def __str__(self):
        return f"{self.student.name} solved {self.problem.title}"
```

1.

Generate Migrations:

```
pipenv run python manage.py makemigrations
pipenv run python manage.py migrate
```

2.

Step 4: Register Models in the Admin Panel

Edit `admin.py` in `contest_app`:

```
from django.contrib import admin
from .models import Student, Problem, Solution

@admin.register(Student)
class StudentAdmin(admin.ModelAdmin):
    list_display = ('name', 'email', 'registration_date')
    search_fields = ('name', 'email')

@admin.register(Problem)
class ProblemAdmin(admin.ModelAdmin):
    list_display = ('title', 'difficulty')

@admin.register(Solution)
class SolutionAdmin(admin.ModelAdmin):
    list_display = ('student', 'problem', 'solved_on')
```

1.

Run the Development Server and log in to the admin panel:

```
pipenv run python manage.py createsuperuser
pipenv run python manage.py runserver
```

2. Open `http://127.0.0.1:8000/admin` and use the superuser credentials to log in.

Step 5: Create the Home Page

1. Create a View

- **File**: `contest_app/views.py`

Code:
python
CopyEdit
```
from django.shortcuts import render

def home(request):
    return render(request, 'home.html', {'title': 'Contestants
Project'})
```

-

---

2. Map the URL

- **File**: `<project_name>/urls.py`

Code:
python
CopyEdit
```
from django.contrib import admin
from django.urls import path
from contest_app import views

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', views.home, name='home'),
]
```

-

---

3. Create the Template

- **File**: `contest_app/templates/home.html`
    - Create the `templates` folder inside the `contest_app` directory if it doesn't exist.

Code:
html

CopyEdit

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>{{ title }}</title>
</head>
<body>
    <h1>Welcome to the Contestants' Problem-Solution Project</h1>
    <p>This project allows students to share and explore problem
solutions.</p>
</body>
</html>
```

-

---

4. Test the Home Page

Start the development server:
bash
CopyEdit

```
pipenv run python manage.py runserver
```

-
- Open your browser and visit:
  `http://127.0.0.1:8000`

You should see the home page with the welcome message and project description.